INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, VOL. 40, 2717–2737 (1997)

AN ALGEBRAICALLY PARTITIONED FETI METHOD FOR PARALLEL STRUCTURAL ANALYSIS: ALGORITHM DESCRIPTION

K. C. PARK, MANOEL R. JUSTINO, JR. AND CARLOS A. FELIPPA

Center for Aerospace Structures and Department of Aerospace Engineering Sciences, Campus Box 429, University of Colorado, Boulder, CO 80309-429, U.S.A.

SUMMARY

An algebraically partitioned FETI method for the solution of structural engineering problems on parallel computers is presented. The present algorithm consists of three attributes: an explicit generation of the orthogonal null-space matrix associated with the interface nodal forces, the floating subdomain rigid-body modes computed from the subdomain static equilibrium equation of the classical force method and the identification of redundant interface force constraint operator that emanates when the interface force computations are localized. Comparisons of the present method with the previously developed differentially partitioned FETI method are offered in terms of the saddle-point formulations at the end of the paper. A companion paper reports implementation details and numerical performance of the proposed algorithm. © 1997 by John Wiley & Sons, Ltd.

KEY WORDS: parallel computation; algebraic partitioning; domain decomposition; structural analysis; finite element method; iterative solution

1. INTRODUCTION

Solution of structural mechanics problems on parallel computers offers a new potential for a quantum jump in solution resolutions, modelling scales and timeliness. The developments of computational procedures for an effective use of parallel computers present new challenges heretofore not encountered in the use of sequential computers. For parallel computational procedures to take a full advantage of the emerging computer architectures, several non-numerical aspects must be incorporated in the parallel solution process.

First, the discrete structural equilibrium equations, which are often generated via the finite element method, need to be partitioned into several substructures, each of whose interior problems are then solved in each processor. This demands an efficient structural problem-specific application of domain decomposition techniques (e.g. see References 1 and 2). Second, the interface forces effected by substructural partitioning must be solved, which satisfy the global assembled equilibrium condition. Ideally, this step needs to be carried out with a minimal interprocessor communication overhead. Third, the solution variables obtained from individual processors must be easily available for subsequent analysis/design activities.

Of numerous developments for solving structural mechanics problems on parallel computers, the finite element tearing and interconnecting (FETI) method is perhaps the most mature[3] to date. For example, the method has been applied to several non-trivial production-level problems and has shown its scalability, an important algorithmic feature. Being a relatively new method, it has been constantly undergoing enhancements. For example, it has been observed in numerical

CCC 0029–5981/97/152717–21$17.50

Received 9 April 1996
Revised 14 November 1996

experiments that additional redundancies in its interface constraint conditions seem to reduce the iteration numbers. In addition, the injection of the so-called troublesome modes also improves its iteration performance, especially for plate and shells. Third, the accuracy loss associated with the subdomain rigid-body modes obtained from the null space of the subdomain stiffness matrices can adversely impact the solution accuracy.

In views of the two formulation and solution paradigms for partitioned solution procedures, viz., *differential partitioning* and *algebraic partitioning*,[4, 5] the FETI method would fall into a differential partitioning category. Specifically, the FETI method is based on the *differentially partitioned* saddle-point framework, with an indefinite partitioned stiffness operators governing the structural internal energy description even though the global assembled stiffness matrix becomes positive definite. For this reason, we will designate the FETI method of Farhat and Roux[3] as a *differentially partitioned* FETI method.

It was pointed out in our previous studies[6, 7] that the differential and algebraic partitioning possess their distinct algorithmic advantages as well as disadvantages in the context of sequential staggered solution procedures. However, their relative merits as applied to parallel computations have not been carefully studied. This has motivated the present authors to formulate an algebraical partitioning procedure as an alternative to the *differentially partitioned* FETI method in the hope of gaining further insight into the some of the algorithmic aspects of both the differential and algebraic partitioned parallel solution procedures while preserving the principal property of scalability.

Specifically, the present *algebraic partitioning* exploits the way the global structural stiffness matrix $\mathbf{K}_{\mathrm{g}}$ is assembled and partitions it into triple product matrices:

$$\mathbf{K}_{\mathrm{g}} = [\mathbf{L}^{\mathrm{T}}][\mathbf{K}^{(s)}][\mathbf{L}], \quad \mathbf{K}^{(s)} = \begin{bmatrix} \mathbf{K}^{(1)} & & & \\ & \mathbf{K}^{(2)} & & \\ & & \ddots & \\ & & & \mathbf{K}^{(n_s)} \end{bmatrix}$$

so that the solution or its inversion is then carried out in three steps:

(a) solve for $\mathbf{p}^{(s)}$ from $[\mathbf{L}^{\mathrm{T}}]\ \mathbf{p}^{(s)} = \mathbf{f}_{\mathrm{g}}$,
(b) solve for $\mathbf{u}^{(s)}$ from $[\mathbf{K}^{(s)}]\ \mathbf{u}^{(s)} = \mathbf{p}^{(s)}$,
(c) solve for $\mathbf{u}_{\mathrm{g}}$ from $[\mathbf{L}]\ \mathbf{u}_{\mathrm{g}} = \mathbf{u}^{(s)}$,

where $[\mathbf{L}]$ is the finite element assembly Boolean matrix, $\mathbf{p}^{(s)}$ is the domain-by-domain internal force vector, $\mathbf{u}^{(s)}$ is the domain-by-domain displacement vector, $[\mathbf{K}^{(s)}]$ is the domain-by-domain substructural stiffness matrices, $\mathbf{f}_{\mathrm{g}}$ is the applied force, $\mathbf{u}_{\mathrm{g}}$ is the displacement of the assembled global system, and the superscript s denotes the subdomain ($s = 1, 2, 3, \ldots, n_s$) where n_s is the total number of subdomains.

First, since the row size of $[\mathbf{L}]^{\mathrm{T}}$ is in general larger than its column size, the solution of $\mathbf{p}^{(s)}$ must be augmented by a null-space basis of $\mathbf{L}$. Second, the solution of $\mathbf{u}^{(s)}$ must also be complemented with a null-space basis of $[\mathbf{K}^{(s)}]$ for each of the floating subdomains. Third, the inevitable relaxation of the kinematic relation, as given in step (c) outlined above, due to subdomain-by-subdomain parallel computations gives rise to an additional constraint for treating localized interface force redundancies. The present paper addresses these three algorithmic attributes, which we derive solely from kinematical considerations. To this end, the paper is organized as follows.

Section 2 reviews the basic differentially partitioned FETI method as presented in Reference 3 as a guide and comparison source for the present algorithm based on algebraic partitioning. The algebraic partitioning is then presented in Section 3. The description of an explicit construction of

the orthogonal null-space matrix associated with the interface nodal forces is presented in Section 4. A comparison with the interface constraint condition of the differentially partitioned FETI method is discussed therein.

Section 5 presents the well-known concept of static equilibrium, which is utilized to derive the rigid-body modes of the floating subdomains. This procedure bypasses the error-prone null-space extraction process from the subdomain stiffness matrices. Section 6 offers two algorithms for constructing the localization operators for computing the interface forces. In addition, the necessary constraint conditions for treating the redundant localized interface forces are also derived.

Section 7 presents algorithmic equivalence between the two present formalisms, viz., the global and the localized procedures, an important algorithmic property of the present algebraic partitioning. Section 8 presents a comparison of the present method with the differentially partitioned FETI method along with discussions. Finally, discussions regarding the present algebraically partitioned FETI methods are offered in Section 9. We should note that implementation aspects and numerical evaluations of the present algorithms are detailed in a companion paper.[8]

2. REVIEW OF THE DIFFERENTIALLY PARTITIONED FETI METHOD

The starting point in the derivation of the FETI method is the substructural energy $J^{(s)}$:

$$J^{(s)} = \mathbf{u}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)} - \tfrac{1}{2}\mathbf{u}^{(s)^{\mathrm{T}}}\mathbf{K}^{(s)}\mathbf{u}^{(s)}, \quad s = 1, 2, \ldots, n_s \tag{1}$$

where $\mathbf{u}^{(s)}$ and $\mathbf{f}^{(s)}$ are the substructure-level displacement and nodal force vectors, $\mathbf{K}^{(s)}$ is the substructural stiffness matrix, the superscript (s) denotes the substructure domain, and n_s is the total number of substructures partitioned.

In order for the substructure-level displacement to yield the desired global displacement, $\mathbf{u}^{(s)}$ must satisfy the following interface condition:

$$\sum_{s=1}^{n_s} \mathbf{B}^{(s)}\mathbf{u}^{(s)} = 0 \tag{2}$$

which states that the interface displacements obtained in the adjacent domains by a partitioned solution procedure must be the same. Hence, $\mathbf{B}^{(s)}$ is the interface displacement compatibility operator consisting of 1, -1 and 0. The energy expression for the global structure is just the sum of substructure-level energy with the constraint condition (2) augmented via Lagrange's multipliers as shown below:

$$J_{\text{total}} = \sum_{s=1}^{n_s} J^{(s)} - \boldsymbol{\lambda}_{\mathrm{b}}^{\mathrm{T}} \sum_{s=1}^{n_s} \mathbf{B}^{(s)}\mathbf{u}^{(s)} \tag{3}$$

It is observed that the constraint equation (2) produces no work. The stationarity of (3) yields the following substructure-level governing equation:

$$\begin{aligned} \mathbf{K}^{(s)}\mathbf{u}^{(s)} &= \mathbf{f}^{(s)} - \mathbf{B}^{(s)^{\mathrm{T}}}\boldsymbol{\lambda}_{\mathrm{b}}, \quad s = 1, 2, \ldots, n_s \\ \sum_{s=1}^{n_s} \mathbf{B}^{(s)}\mathbf{u}^{(s)} &= 0 \end{aligned} \tag{4}$$

which can be written in matrix form as

$$\begin{bmatrix} \mathbf{K}^{(1)} & \cdot & \cdot & \cdots & \mathbf{B}^{(1)^{\mathrm{T}}} \\ \cdot & \mathbf{K}^{(2)} & \cdot & \cdots & \mathbf{B}^{(2)^{\mathrm{T}}} \\ \cdot & \cdot & \mathbf{K}^{(3)} & \cdots & \mathbf{B}^{(3)^{\mathrm{T}}} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \mathbf{B}^{(1)} & \mathbf{B}^{(2)} & \mathbf{B}^{(3)} & \cdots & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{u}^{(1)} \\ \mathbf{u}^{(2)} \\ \mathbf{u}^{(3)} \\ \vdots \\ \boldsymbol{\lambda}_{\mathrm{b}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^{(1)} \\ \mathbf{f}^{(2)} \\ \mathbf{f}^{(3)} \\ \vdots \\ 0 \end{Bmatrix} \tag{5}$$

Observe that, whereas (4a) is strictly local for each substructure provided $\boldsymbol{\lambda}_{\mathrm{b}}$ is known, the constraint equation (4b) extends over several subdomains that are connected. Nevertheless, the substructural displacement vector $\mathbf{u}^{(s)}$ is of local nature. This suggests a possibility of non-unique construction of the interface displacement compatibility operator $\mathbf{B}^{(s)}$. This is because, as long as the product $\mathbf{B}^{(s)^{\mathrm{T}}}\boldsymbol{\lambda}_{\mathrm{b}}$ is unique, a variety of combinations of different $\mathbf{B}^{(s)}$ and $\boldsymbol{\lambda}_{\mathrm{b}}$ may satisfy the product to be unique. This has algorithmic ramifications as will be discussed later.

The differentially partitioned FETI method begins with the solution of $\mathbf{u}^{(s)}$ from (4a):

$$\mathbf{u}^{(s)} = \mathbf{K}^{(s)^{+}}(\mathbf{f}^{(s)} - \mathbf{B}^{(s)^{\mathrm{T}}}\boldsymbol{\lambda}_{\mathrm{b}}) - \mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \tag{6}$$

where $\mathbf{K}^{(s)^{+}}$ is a generalized inverse of $\mathbf{K}^{(s)}$ and $\mathbf{R}^{(s)}$ is the null-space matrix satisfying

$$\mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{K}^{(s)}\mathbf{u}^{(s)} = 0 \tag{7}$$

The Lagrange multipliers $\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}$ denote the complementary displacement vector that accounts for the rigid-body motions for floating substructures. Thus, the solution of $\mathbf{u}^{(s)}$ by the FETI method is reduced to the solution of the interface force $\boldsymbol{\lambda}_{\mathrm{b}}$ and the complementary displacement vector $\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}$.

In order to obtain the appropriate equations for the two Lagrange multipliers, we substitute (4a) into (7) and (6) into (4b) to arrive at

$$\begin{bmatrix} \mathbf{F}_{\mathrm{b}} & \mathbf{G} \\ \mathbf{G}^{\mathrm{T}} & 0 \end{bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_{\mathrm{b}} \\ \boldsymbol{\lambda}_{\mathrm{r}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b}_{\mathrm{b}} \\ \mathbf{b}_{\mathrm{r}} \end{Bmatrix} \tag{8}$$

where

$$\begin{aligned} \mathbf{F}_{\mathrm{b}} &= \sum_{s=1}^{n_s} \mathbf{B}^{(s)}\mathbf{K}^{(s)^{+}}\mathbf{B}^{(s)^{\mathrm{T}}} \\ \mathbf{G} &= \{\mathbf{B}^{(1)}\mathbf{R}^{(1)} \ldots \mathbf{B}^{(N_s)}\mathbf{R}^{(N_s)}\} \\ \boldsymbol{\lambda}_{\mathrm{r}} &= \{\boldsymbol{\lambda}_{\mathrm{r}}^{(1)} \ldots \boldsymbol{\lambda}_{\mathrm{r}}^{(n_s)}\}^{\mathrm{T}} \\ \mathbf{b}_{\mathrm{b}} &= \sum_{s=1}^{n_s} \mathbf{B}^{(s)}\mathbf{K}^{(s)^{+}}\mathbf{f}^{(s)} \\ \mathbf{b}_{\mathrm{r}} &= \{\mathbf{f}^{(s)^{\mathrm{T}}}\mathbf{R}^{(s)}\}^{\mathrm{T}}, \quad s = 1, 2, \ldots, n_s \end{aligned} \tag{9}$$

The foregoing review of the *differentially partitioned* FETI method reveals that its performance is dictated by the three algorithmic aspects: the choice of the interface displacement compatibility operator $\mathbf{B}^{(s)}$, the accuracy of the null-space matrix $\mathbf{R}^{(s)}$, and the so-called troublesome modes that can accelerate iteration convergence. Much is now known about the three algorithmic aspects since its first development in 1990. For example, $\mathbf{F}_{\mathrm{b}}$ becomes singular whenever the interface operator $\mathbf{B}^{(s)}$ possesses redundancies when the interface nodes become cross points, i.e. a node shared by

more than two substructures. Such redundancies are shown to accelerate the conjugate gradient iterations.

3. ALGEBRAIC PARTITIONING OF GLOBAL DISCRETE EQUATIONS

In contrast to the differentially partitioned FETI method reviewed above, the present algebraic partitioning starts with the assembled finite element discrete equilibrium equation whose solution matrix is given in a triple-matrix factored form:

$$\mathbf{L}^{\mathrm{T}}\mathbf{K}^{(s)}\mathbf{L}\mathbf{u}_{\mathrm{g}} = \mathbf{f}_{\mathrm{g}}, \quad \mathbf{K}^{(s)} = \begin{bmatrix} \mathbf{K}^{(1)} & & & \\ & \mathbf{K}^{(2)} & & \\ & & \ddots & \\ & & & \mathbf{K}^{(n_s)} \end{bmatrix} \tag{10}$$

where $\mathbf{L}$ is the assembly Boolean matrix that facilitates the coupling of the subdomains, $\mathbf{K}^{(s)}$ is the subdomain-by-subdomain stiffness matrix, $\mathbf{u}_{\mathrm{g}}$ is the global nodal displacement vector, and $\mathbf{f}_{\mathrm{g}}$ is the external force vector, respectively. It should be noted that $\mathbf{L}$ is uniquely given once the finite element discretization is complete for the entire structure.

In order to effect the three solution steps outlined in Introduction, we employ two important kinematical and physical relations:

Domain-by-domain vs. Global displacement relation:

$$\mathbf{u}^{(s)} - \mathbf{L}\mathbf{u}_{\mathrm{g}} = 0 \tag{11}$$

Domain-by-domain internal reaction force:

$$\mathbf{p}^{(s)} = \mathbf{K}^{(s)}\mathbf{u}^{(s)} \tag{12}$$

Note that (11) is simply an algebraic statement that the substructural displacement $\mathbf{u}^{(s)}$ is related to the assembled global displacement $\mathbf{u}_{\mathrm{g}}$ by the assembly Boolean matrix $\mathbf{L}$. We now outline the present algebraically partitioned parallel FETI method below.

Step 1: Algebraic partitioning of the global domain into subdomains. This step simply involves the algebraic domain decomposition of $\mathbf{L}^{\mathrm{T}}$, i.e. the solution of

$$\mathbf{L}^{\mathrm{T}}\mathbf{p}^{(s)} = \mathbf{f}_{\mathrm{g}} \tag{13}$$

For clarity of presentation purposes, we partition $\mathbf{p}^{(s)}$ into two parts: those corresponding to the interior nodal degrees of freedom for all the subdomains and those belonging to the subdomain interfaces:

$$\begin{gathered} \mathbf{p}^{(s)} = \begin{Bmatrix} \mathbf{p}_{\mathrm{i}}^{(s)} \\ \mathbf{p}_{\mathrm{b}}^{(s)} \end{Bmatrix}, \quad \mathbf{u}^{(s)} = \begin{Bmatrix} \mathbf{u}_{\mathrm{i}}^{(s)} \\ \mathbf{u}_{\mathrm{b}}^{(s)} \end{Bmatrix}, \quad \mathbf{u}_{\mathrm{g}} = \begin{Bmatrix} \mathbf{u}_{\mathrm{gi}} \\ \mathbf{u}_{\mathrm{gb}} \end{Bmatrix}, \quad \mathbf{f}_{\mathrm{g}} = \begin{Bmatrix} \mathbf{f}_{\mathrm{gi}} \\ \mathbf{f}_{\mathrm{gb}} \end{Bmatrix} \\ \mathbf{L} = \begin{bmatrix} \mathbf{I}_{\mathrm{i}} & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{\mathrm{b}} \end{bmatrix}, \quad \begin{Bmatrix} \mathbf{f}_{\mathrm{i}}^{(s)} \\ \mathbf{f}_{\mathrm{b}}^{(s)} \end{Bmatrix} = \begin{bmatrix} \mathbf{I}_{\mathrm{i}} & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{\mathrm{b}}^{\mathrm{T}+} \end{bmatrix} \mathbf{f}_{\mathrm{g}} \end{gathered} \tag{14}$$

where $\mathbf{I}_{\mathrm{i}}$ is the identity matrix that accounts for all of the subdomain interior nodes, $\mathbf{L}_{\mathrm{b}}$ is the interface boundary nodal connectivity matrix, $\mathbf{L}_{\mathrm{b}}^{\mathrm{T}+}$ is a generalized inverse of $\mathbf{L}_{\mathrm{b}}^{\mathrm{T}}$, and $\mathbf{f}_{\mathrm{i}}^{(s)}$ and $\mathbf{f}_{\mathrm{b}}^{(s)}$ are the applied forces acting on the substructural interior and boundary nodes, respectively.

Solving for the subdomain reaction force vector $\mathbf{p}^{(s)}$ yields

$$\mathbf{p}^{(s)} = \mathbf{K}^{(s)}\mathbf{L}\mathbf{u}_{\mathrm{g}} = \begin{Bmatrix} \mathbf{f}_{\mathrm{i}}^{(s)} \\ \mathbf{f}_{\mathrm{b}}^{(s)} \end{Bmatrix} - \begin{Bmatrix} \mathbf{0} \\ \mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}} \end{Bmatrix} \tag{15}$$

where $\mathbf{N}_{\mathrm{b}}$ is a null-space basis of $\mathbf{L}_{\mathrm{b}}^{\mathrm{T}}$, and $\boldsymbol{\lambda}_{\mathrm{b}}$ is the complementary contribution to the solution of $\mathbf{p}^{(s)}$ due to the algebraic partitioning. From the physical point of view, the null-space matrix $\mathbf{N}_{\mathrm{b}}$ is *the displacement compatibility operator* that satisfies the following condition:

$$\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{u}_{\mathrm{b}}^{(s)} = \mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{L}_{\mathrm{b}}\mathbf{u}_{\mathrm{gb}} = \mathbf{0} \tag{16}$$

Observe that the Lagrange multiplier vector $\boldsymbol{\lambda}_{\mathrm{b}}$ represents a generalized domain-to-domain interface force vector. Therefore, the present algebraic partitioning step for the solution of $\mathbf{p}^{(s)}$ consists of obtaining the two matrices, $\mathbf{L}_{\mathrm{b}}^{\mathrm{T}^+}$ and $\mathbf{N}_{\mathrm{b}}$.

It turns out the null-space matrix $\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}$ is a global matrix, viz., computations of the associated interface force $\boldsymbol{\lambda}_{\mathrm{b}}$ cannot be carried out independently in parallel. Hence, in order to realize their localized subdomain-by-subdomain computations, the displacement compatibility operator $\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}$ needs to be localized to an uncoupled form $\mathbf{N}_{\mathrm{b}}^{(s)}$. This results in a relaxation of the constraint equation (16) to

$$\mathbf{N}^{(s)^{\mathrm{T}}}{}_{\mathrm{b}}\mathbf{u}^{(s)} - \mathbf{Z}_{\lambda}\mathbf{u}_{\mathrm{gb}} = \mathbf{0}, \quad \mathbf{Z}_{\lambda} = \mathbf{N}^{(s)^{\mathrm{T}}}{}_{\mathrm{b}}\mathbf{L}_{\mathrm{b}} \tag{17}$$

where $\mathbf{u}_{\mathrm{gb}}$ corresponds to the global displacement at the substructural boundary nodes, and $\mathbf{Z}_{\lambda}$ is the constraint operator on the redundant localized interface force $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$ that must satisfy

$$\mathbf{Z}_{\lambda}^{\mathrm{T}}\boldsymbol{\lambda}_{\mathrm{b}}^{(s)} = 0 \tag{18}$$

Remark 1. A simplest choice of the localized compatibility operator $\mathbf{N}_{\mathrm{b}}^{(s)}$ would be an identity matrix given by

$$\mathbf{N}_{\mathrm{b}}^{(s)} = \mathbf{I}_{\mathrm{b}} \Rightarrow \mathbf{Z}_{\lambda} = \mathbf{L}_{\mathrm{b}} \tag{19}$$

which corresponds to treating the boundary interface force $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$ at each interface node to be independent. This choice leads to computational simplicity as well as software modularity. This aspect is further discussed in Sections 6 and 7.

Step 2: Subdomain-by-subdomain solution. Assuming that the subdomain reaction force vector $\mathbf{p}^{(s)}$ is obtained in Step 1 given by (15), one can solve for the subdomain displacement vector given by

$$\mathbf{K}^{(s)}\mathbf{u}^{(s)} = \mathbf{p}^{(s)} \tag{20}$$

Solving for $\mathbf{u}^{(s)}$ from the internal equilibrium equation (20) yields

$$\mathbf{u}^{(s)} = \mathbf{F}^{(s)}\mathbf{p}^{(s)} - \mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \tag{21}$$

where $\mathbf{R}^{(s)}$ is the subdomain-by-subdomain rigid-body modes, $\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}$ is the subdomain displacement vector that accounts for the rigid-mode motions among the floating substructures, and $\mathbf{F}^{(s)}$ is the subdomain flexibility matrix that satisfies the following property:

$$\mathbf{K}^{(s)}\mathbf{F}^{(s)} = \mathbf{F}^{(s)}\mathbf{K}^{(s)} = \mathbf{I} - \mathbf{R}^{(s)}(\mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{R}^{(s)})^{-1}\mathbf{R}^{(s)^{\mathrm{T}}} \tag{22}$$

Observe that the internal reaction force $\mathbf{p}^{(s)}$ must satisfy the well-known static equilibrium condition:

$$\mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{p}^{(s)} = \mathbf{0} \tag{23}$$

For this reason $\mathbf{R}^{(s)^{\mathrm{T}}}$ will be called a *subdomain equilibrium operator*.

Substituting $\mathbf{p}^{(s)}$ from (15) into (21) leads to

$$\mathbf{u}^{(s)} = \mathbf{L}\mathbf{u}_{\mathrm{g}} = \mathbf{F}^{(s)}\left\{\mathbf{f}^{(s)} - \begin{Bmatrix}\mathbf{0}\\ \mathbf{N}_{\mathrm{b}}\end{Bmatrix}\boldsymbol{\lambda}_{\mathrm{b}}\right\} - \mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \tag{24}$$

Numerically, $\mathbf{R}^{(s)}$ is a null-space basis of $\mathbf{K}^{(s)}$ or commonly referred to as subdomain rigid-body modes. It will be shown that the present view of $\mathbf{R}^{(s)}$ as the *subdomain static equilibrium operator* leads to an algorithm for constructing it purely from kinematical considerations instead of extracting it from $\mathbf{K}^{(s)}$.

Step 3: Solution of $\boldsymbol{\lambda}_{\mathrm{b}}$ *and* $\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}$. The two solution steps outlined in the preceding sections can be brought together to form a coupled difference equation. As the two choices of the displacement compatibility conditions, viz., (16) and (17), lead to two different solution strategies, we describe them in two separate subsections. It turns out the condition (16) leads to a global interface force computation algorithm, whereas the localized condition (17) to a localized interface computation.

Case 1: Global computing of $\boldsymbol{\lambda}_{\mathrm{b}}$. First, we impose the subdomain equilibrium condition (23) to the subdomain reaction force vector (15) to obtain

$$\mathbf{R}^{(s)^{\mathrm{T}}}(\mathbf{f}^{(s)} - \mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}}) = 0 \tag{25}$$

where for notational simplicity we used $\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}$ to imply $\langle \mathbf{0}\ \mathbf{N}_{\mathrm{b}}\rangle^{\mathrm{T}}$ in an interchangeable manner.

Second, we apply the subdomain displacement compatibility conditions (16)–(24) to obtain

$$\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\{\mathbf{F}^{(s)}(\mathbf{f}^{(s)} - \mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}}) - \mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}\} = 0 \tag{26}$$

Combining (25) and (26) we arrive at an equation set for computing the global interface force $\boldsymbol{\lambda}_{\mathrm{b}}$:

$$\begin{bmatrix}\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{F}^{(s)}\mathbf{N}_{\mathrm{b}} & \mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{R}^{(s)}\\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{N}_{\mathrm{b}} & \mathbf{0}\end{bmatrix}\begin{Bmatrix}\boldsymbol{\lambda}_{\mathrm{b}}\\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)}\end{Bmatrix} = \begin{Bmatrix}\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{F}^{(s)}\mathbf{f}^{(s)}\\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)}\end{Bmatrix} \tag{27}$$

Comparing the above equation (27) with the differentially partitioned FETI method (8), we conclude that the present global method has the same form of the FETI method, with one exception. That is, the present global interface force vector $\boldsymbol{\lambda}_{\mathrm{b}}$ consists of a linearly independent set. Also, the solution matrix has a full rank. This may open new possibilities for solving the interface problems (27).

Case 2: Localized Computing of $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$. The bulk of computations required for the solution of $\boldsymbol{\lambda}_{\mathrm{b}}$ and $\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}$ from the global interface equation (27) is the term $\mathbf{N}_{\mathrm{b}}^{\mathrm{T}}\mathbf{F}^{(s)}\mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}}$. Hence, if it is to be efficient for parallel computations, this term should be computed in each processor without any interprocessor communication. In other words, its computations need to be *localized*. To this end, we employ the relaxed interface displacement compatibility condition (17) and substitute (24) for $\mathbf{u}^{(s)}$ to yield

$$\mathbf{N}^{(s)^{\mathrm{T}}}{}_{\mathrm{b}}\{\mathbf{F}^{(s)}(\mathbf{f}^{(s)} - \mathbf{N}_{\mathrm{b}}^{(s)}\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}) - \mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}^{(s)}\} - \mathbf{Z}_{\mathcal{A}}\mathbf{u}_{\mathrm{gb}} = \mathbf{0} \tag{28}$$

The present localized equation is obtained by combining (18), the localized equivalent of (25) and (28):

$$\begin{bmatrix} \mathbf{N}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{N}_{\mathrm{b}}^{(s)} & \mathbf{N}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{R}^{(s)} & \mathbf{Z}_{\lambda} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{N}_{\mathrm{b}}^{(s)} & \mathbf{0} & \mathbf{0} \\ \mathbf{Z}_{\lambda}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \\ \mathbf{u}_{\mathrm{gb}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{N}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)} \\ \mathbf{0} \end{Bmatrix} \tag{29}$$

$$\mathbf{F}^{(s)} = \begin{bmatrix} \mathbf{F}^{(1)} & & & \\ & \mathbf{F}^{(2)} & & \\ & & \ddots & \\ & & & \mathbf{F}^{(n_s)} \end{bmatrix}, \quad \mathbf{N}_{\mathrm{b}}^{(s)} = \begin{bmatrix} \mathbf{N}_{\mathrm{b}}^{(1)} \\ \mathbf{N}_{\mathrm{b}}^{(2)} \\ \vdots \\ \mathbf{N}_{\mathrm{b}}^{(n_s)} \end{bmatrix}$$

where we have used a localized form of the subdomain equilibrium condition (23) analogous to the global case (25).

In passing we note that Farhat and Mandel[9] recently reported a two-level FETI method in which they introduce their 'corner modes' operator $\mathbf{C}$ with the following property:

$$\mathbf{C}_{\mathrm{w}}^{\mathrm{T}}\mathbf{B}^{(s)^{\mathrm{T}}}\mathbf{w}^{(s)} = \mathbf{0} \tag{30}$$

Therefore, their corner modes *explicitly* enforce the transverse interface displacement continuity. On the other hand, the present localization interface condition (17) constrains the redundant interface forces. In other words, the present relaxed compatibility condition (17) is a force constraint, whereas the constraint employed in their study is a displacement constraint. Further study of their interrelationships, viz., $\mathbf{B}^{(s)}$ vs. $\mathbf{N}_{\mathrm{b}}^{(s)}$ and $\mathbf{Z}_{\lambda}$ vs. the 'corner modes' $\mathbf{C}_{\mathrm{w}}^{\mathrm{T}}$, may shed some new insight for improving iterative structural analysis methods on parallel computers.

Step 4: Solution of the global equations from the subdomain solutions. The solution vector of the global system $\mathbf{u}_{\mathrm{g}}$ can be obtained by a least-squares projection of the domain-by-domain solution $\mathbf{u}^{(s)}$ obtained in Step 2. This is accomplished from (11) as

$$\begin{aligned} \mathbf{u}_{\mathrm{g}} &= \mathbf{L}^{+}\mathbf{u}^{(s)} \\ &= \mathbf{L}^{+}\mathbf{F}^{(s)}\{\mathbf{f}^{(s)} - \mathbf{N}_{\mathrm{b}}^{(s)}\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}\} - \mathbf{L}^{+}\mathbf{R}^{(s)}\boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \end{aligned} \tag{31}$$

where $\mathbf{N}_{\mathrm{b}}$ and $\boldsymbol{\lambda}_{\mathrm{b}}$ can be either the global quantities obtained from (27) or the localized quantities obtained from (29).

We now present a procedure for deriving the global displacement compatibility operator $\mathbf{N}_{\mathrm{b}}$, the static equilibrium operator $\mathbf{R}^{(s)}$, the localized displacement compatibility operator $\mathbf{N}_{\mathrm{b}}^{(s)}$, and the localized interface constraint operator $\mathbf{Z}_{\lambda}$. It will be shown that these matrices are obtained solely from kinematical considerations.

4. COMPUTATION OF THE GLOBAL DISPLACEMENT COMPATIBILITY OPERATOR $\mathbf{N}_{\mathrm{b}}$

We will first show the computation of $\mathbf{N}_{\mathrm{b}}$ with a four-element plane beam structure shown in Figure 1. For this example, we partition it into four subdomains. The assembly Boolean matrix that relates the global displacement vector $\mathbf{u}_{\mathrm{g}}$ to the subdomain displacement vector $\mathbf{u}^{(s)}$ is given

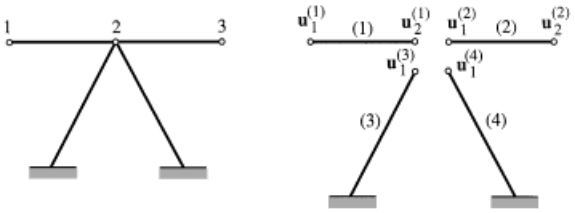

Figure 1. Four-subdomain geometry

by

$$\mathbf{u}^{(s)} = \mathbf{L}\mathbf{u}_{\mathrm{g}}, \quad \mathbf{u}^{(s)} = \begin{Bmatrix} \mathbf{u}_1^{(1)} \\ \mathbf{u}_2^{(1)} \\ \mathbf{u}_1^{(2)} \\ \mathbf{u}_2^{(2)} \\ \mathbf{u}^{(3)} \\ \mathbf{u}^{(4)} \end{Bmatrix}, \quad \mathbf{u}_{\mathrm{g}} = \begin{Bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{Bmatrix}, \quad \mathbf{u}_{\mathrm{i}} = \begin{Bmatrix} u_x \\ u_y \\ \theta_z \end{Bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 \\ 0 & \mathbf{I} & 0 \end{bmatrix} \tag{32}$$

It should be emphasized that the above finite element assembly operator $\mathbf{L}$ is unique. Since $\mathbf{L}$ is a (18×9) matrix and has a column rank of 9, the null-space basis matrix $\mathbf{N}_{\mathrm{b}}$ is a (18×9) matrix with its column rank of 9.

For this example problem, the null-space matrix of $\mathbf{L}$ is obtained by taking the compacted second column, viz.,

$$\bar{\mathbf{L}}\ (:,2) = \begin{Bmatrix} \mathbf{I} \\ \mathbf{I} \\ \mathbf{I} \\ \mathbf{I} \end{Bmatrix} \tag{33}$$

where we have used a Matlab convention for extracting submatrices. The null space of the above column matrix is given by

$$\bar{\mathbf{N}}_{\mathrm{b}} = \begin{Bmatrix} \mathbf{I} & 0 & \mathbf{I} \\ -\mathbf{I} & 0 & \mathbf{I} \\ 0 & \mathbf{I} & -\mathbf{I} \\ 0 & -\mathbf{I} & -\mathbf{I} \end{Bmatrix} \tag{34}$$

Expanding the compact form of the above null-space matrix into the full-profile matrix, we obtain

$$\mathbf{N}_{\mathrm{b}} = \begin{Bmatrix} 0 & 0 & 0 \\ \mathbf{I} & 0 & \mathbf{I} \\ -\mathbf{I} & 0 & \mathbf{I} \\ 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{I} \\ 0 & -\mathbf{I} & -\mathbf{I} \end{Bmatrix} \tag{35}$$

Note that the preceding example computation of the null-space matrix $\mathbf{N}_b$ is carried out *explicitly* on column-by-column operations. Hence, they can be constructed with simple algebra.

As additional examples, a few sample explicit null-space matrices are given below.

For a three-node cross point, we have

$$\bar{\mathbf{L}} = \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \Rightarrow \bar{\mathbf{N}}_b = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ 0 & -2 \end{bmatrix} \tag{36}$$

and for a five-node cross point, $\bar{\mathbf{N}}_b$ is given by

$$\bar{\mathbf{L}} = \begin{Bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{Bmatrix} \Rightarrow \bar{\mathbf{N}}_b = \begin{bmatrix} 1 & 0 & 1 & 1 \\ -1 & 0 & 1 & 1 \\ 0 & 1 & -1 & 1 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & -2 \end{bmatrix} \tag{37}$$

To summarize, the present displacement compatibility operator $\mathbf{N}_b$ is computed from the finite element assembly Boolean matrix $\mathbf{L}$ which is unique. The null-space extraction is carried out on the compacted column-by-column vectors. In addition, the explicit null-space formulas developed herein yield an orthogonal $\mathbf{N}_b$ matrix, thus yielding the subdomain displacement compatibility operator $\mathbf{N}_b$ to consist of linearly independent basis vectors.

In practice, each column is normalized to form an orthonormal matrix, which is shown to offer beneficial effects on the condition number of the resulting preconditioned iteration matrix. Note also that each column constitutes an orthogonal basis, a contrast to $\mathbf{B}^{(s)}$ matrix used in the differential FETI method (2).

5. COMPUTATION OF SUBDOMAIN EQUILIBRIUM OPERATOR $\mathbf{R}^{(s)}$

This section presents two methods of computing the subdomain equilibrium operator $\mathbf{R}^{(s)}$. It should be recalled that this operator is physically subdomain rigid-body modes. This distinction, namely, the subdomain equilibrium operator vs. the subdomain rigid-body modes, although referring to the same physical behaviour, leads to two distinct algorithms for constructing $\mathbf{R}^{(s)}$.

If one views $\mathbf{R}^{(s)}$ as a null-space matrix of the subdomain stiffness matrix $\mathbf{K}^{(s)}$, $\mathbf{R}^{(s)}$ must be obtained from $\mathbf{K}^{(s)}$ whenever it becomes singular. On the other hand, by taking the present approach of regarding it as the subdomain equilibrium operator, its construction algorithm can be purely kinematic. It is the latter viewpoint we are adopting.

There is another important consequence that accrues from the present viewpoint. When one views $\mathbf{R}^{(s)}$ simply as rigid-body modes, its major role is to bring the *floating* subdomains together kinematically. However, if we recall the subdomain equilibrium equation (25) below

$$\mathbf{R}^{(s)^{\mathrm{T}}}(\mathbf{f}^{(s)} - \mathbf{N}_b \boldsymbol{\lambda}_b) = 0 \tag{25}$$

it becomes clear that its accuracy dictates the accuracy of the subdomain equilibrium, an important consideration. In other words, the role of $\mathbf{R}^{(s)}$ is far more than just bringing the subdomains back together geometrically; it influences the accuracy of the subdomain equilibrium, consequently the accuracy of the global force residual solution.

We will first offer a review of computing both $\mathbf{K}^{(s)^+}$ and $\mathbf{R}^{(s)}$. We will then show the present approach, first, for elements with both translational and rotational degrees of freedom. The case of elements with only translational degrees of freedom will then follow.

5.1. *Computations of* $\mathbf{R}^{(s)}$ *employed in the differentially partitioned FETI method*

In the differentially partitioned FETI method, the null-space matrix of $\mathbf{K}^{(s)}$ is constructed as follows. For each subdomain s, the rank of $\mathbf{K}^{(s)}$ is determined and $\mathbf{K}^{(s)}$ is rearranged as

$$\mathbf{K}^{(s)} = \begin{bmatrix} \mathbf{K}_{(\mathrm{cc})} & \mathbf{K}_{(\mathrm{cr})} \\ \mathbf{K}_{(\mathrm{rc})} & \mathbf{K}_{(\mathrm{rr})} \end{bmatrix} \tag{38}$$

where $\mathbf{K}_{(\mathrm{cc})}$ is a square matrix with a full rank. Then, a generalized inverse $\mathbf{K}^{(s)^+}$ and $\mathbf{R}^{(s)}$ are computed according to

$$\mathbf{K}^{(s)^+} = \begin{bmatrix} \mathbf{K}_{(\mathrm{cc})}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{R}^{(s)} = \begin{bmatrix} -\mathbf{K}_{(\mathrm{cc})}^{-1}\mathbf{K}_{(\mathrm{cr})} \\ \mathbf{I} \end{bmatrix} \tag{39}$$

We now present the present procedure for deriving the subdomain equilibrium operator $\mathbf{R}^{(s)}$ from the classical static equilibrium considerations.

5.2. *Computation of* $\mathbf{R}^{(s)}$ *for subdomains with translational and rotational degrees of freedom*

Consider a completely free–free domain where three translational and three moments are acting as shown in Figure 2. In order for the domain to be in statical equilibrium, the sum of the forces and moments with respect to a fixed point must vanish. This static equilibrium condition can be written in matrix form as

$$\begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \boldsymbol{\chi}_{(1)} & \mathbf{I}_3 \end{bmatrix} \begin{Bmatrix} \mathbf{Q}_{(1)} \\ \mathbf{M}_{(1)} \end{Bmatrix} + \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \boldsymbol{\chi}_{(2)} & \mathbf{I}_3 \end{bmatrix} \begin{Bmatrix} \mathbf{Q}_{(2)} \\ \mathbf{M}_{(2)} \end{Bmatrix} + \cdots + \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \boldsymbol{\chi}_{(n)} & \mathbf{I}_3 \end{bmatrix} \begin{Bmatrix} \mathbf{Q}_{(n)} \\ \mathbf{M}_{(n)} \end{Bmatrix} = 0 \tag{40}$$

where $\mathbf{I}_3$ is the (3×3)-identity matrix, $\mathbf{Q}_i$ and $\mathbf{M}_i$ designate the three translational force vector and three moments at node i, respectively, and $\boldsymbol{\chi}_{(i)}$ is the well-known skew symmetric matrix given by

$$\boldsymbol{\chi}_{(i)} = \begin{bmatrix} 0 & -(z_i - z_0) & (y_i - y_0) \\ (z_i - z_0) & 0 & -(x_i - x_0) \\ -(y_i - y_0) & (x_i - x_0) & 0 \end{bmatrix} \tag{41}$$

in which (x_i, y_i, z_i) and (x_0, y_0, z_0) are the co-ordinates at node i and the reference node 0, respectively.

Observe that, in view of (23) or (25), the static subdomain equilibrium operator $\mathbf{R}^{(s)}$ can be obtained from (40) as

$$\mathbf{R}^{(s)^{\mathrm{T}}} = [\, \mathbf{r}_1 \;\; \mathbf{r}_2 \;\; \ldots \;\; \mathbf{r}_n \,]$$

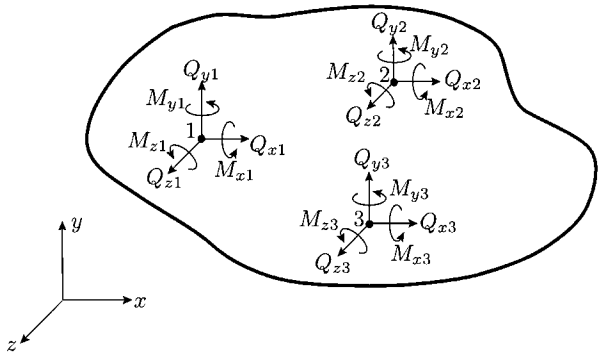

Figure 2. Subdomain with both translational and rotational displacements

$$\mathbf{r}_i = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \boldsymbol{\chi}_{(i)} & \mathbf{I}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -(z_i - z_0) & (y_i - y_0) & 1 & 0 & 0 \\ (z_i - z_0) & 0 & -(x_i - x_0) & 0 & 1 & 0 \\ -(y_i - y_0) & (x_i - x_0) & 0 & 0 & 0 & 1 \end{bmatrix} \tag{42}$$

In the above equation the six rows in $\mathbf{r}_i$ correspond to $(u, v, w, \theta_x, \theta_y, \theta_z)$ rigid modes, where (u, v, w) and $(\theta_x, \theta_y, \theta_z)$ are three translational displacements and three rotations, respectively. Hence, for completely free–free subdomains the computation of subdomain equilibrium operator is a simple task. Let us now consider when the domain is partially constrained.

5.2.1. When u at (x_0, y_0, z_0) *is fixed.* For this case, we have no u-directional rigid mode. Thus, we must eliminate the first row and columns corresponding to the u-mode:

$$\mathbf{r}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\vdots \tag{43}$$

$$\mathbf{r}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -(z_i - z_0) & (y_i - y_0) & 1 & 0 & 0 \\ 0 & -(x_i - x_0) & 0 & 1 & 0 \\ (x_i - x_0) & 0 & 0 & 0 & 1 \end{bmatrix}, \quad i \neq 0$$

5.2.2. When u at (x_0, y_0, z_0) *and* (x_p, y_p, z_p) *are fixed.* For this case not only u-mode disappears but also rotational modes must be constrained out. This is accomplished as follows. We obtain the distance vector from node 0 to node p:

$$\mathbf{d}_{0p} = (x_p - x_0)\mathbf{i} + (y_p - y_0)\mathbf{j} + (z_p - z_0)\mathbf{k} \tag{44}$$

then perform the cross product $\mathbf{d}_{0p} \times \mathbf{i}$ to obtain

$$\mathbf{d}_{0p} \times \mathbf{i} = -(y_p - y_0)\mathbf{k} + (z_p - z_0)\mathbf{j} \tag{45}$$

Depending upon the value of the co-ordinates of $\mathbf{d}_{0p}$, we have the following rotational constraints:

(a) If $(y_p - y_0) = 0$ then $\mathbf{j}$ or θ_y-mode must be constrained;
(b) If $(z_p - z_0) = 0$ then $\mathbf{k}$ or θ_z-mode must be constrained;
(c) Finally, if $(y_p - y_0) \neq 0$ and $(z_p - z_0) \neq 0$, then both θ_y-mode and θ_z-mode must be constrained. Other cases can be similarly dealt with.

5.3. Computation of $\mathbf{R}^{(s)}$ *for subdomains with only translational degrees of freedom*

The preceding algorithm is applicable to all of the finite elements whose nodal degrees of freedom include *rotational* ones. For truss and solid elements which have only three translational ones, one must also construct the rotational rigid modes.

The translational and moment equilibrium equation for a solid can be obtained as follows (see Figure 3):

$$\begin{bmatrix} \mathbf{I}_3 \\ \boldsymbol{\chi}_{(1)} \end{bmatrix} \{\mathbf{Q}_{(1)}\} + \begin{bmatrix} \mathbf{I}_3 \\ \boldsymbol{\chi}_{(2)} \end{bmatrix} \{\mathbf{Q}_{(2)}\} + \cdots + \begin{bmatrix} \mathbf{I}_3 \\ \boldsymbol{\chi}_{(n)} \end{bmatrix} \{\mathbf{Q}_{(n)}\} = 0 \tag{46}$$

where $\mathbf{Q}_i$ designates the three translational force vector at node i, and $\boldsymbol{\chi}_{(i)}$ is given by (41). Therefore, the subdomain equilibrium operator $\mathbf{R}^{(s)}$ becomes

$$\mathbf{R}^{(s)^{\mathrm{T}}} = [\,\mathbf{r}_1 \quad \mathbf{r}_2 \quad \ldots \quad \mathbf{r}_n\,]$$

$$\mathbf{r}_i = \begin{bmatrix} \mathbf{I}_3 \\ \boldsymbol{\chi}_{(i)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -(z_i - z_0) & (y_i - y_0) \\ (z_i - z_0) & 0 & -(x_i - x_0) \\ -(y_i - y_0) & (x_i - x_0) & 0 \end{bmatrix} \tag{47}$$

It should be noted that the procedure for treating the partially constrained boundary conditions follows the same steps offered in the previous section.

Remark 2. The present procedure for obtaining the necessary floating subdomain equilibrium operator is *element-independent*. In other words, higher-order stress/strain enrichments will not affect the outcome as long as the elements under consideration are rank-sufficient.

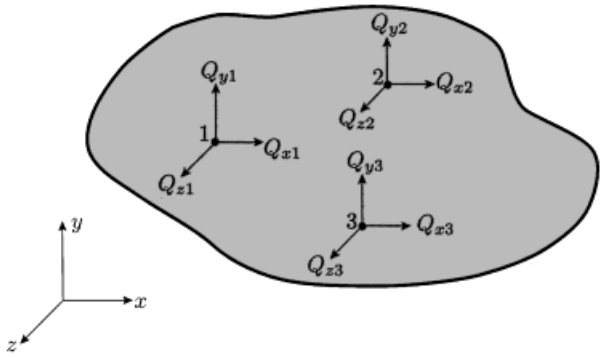

Figure 3. Subdomain with translational displacement only

6. LOCALIZATION OPERATOR $\mathbf{N}_{\mathrm{b}}^{(s)}$ AND INTERFACE FORCE CONSTRAINT OPERATOR $\mathbf{Z}_{\lambda}$

It turns out that there are several possibilities for computing the localized interface force constraint operator $\mathbf{Z}_{\lambda}$ used in (28). We illustrate two possibilities with the plane beam example shown in Figure 1. Note that, for this example case, the interface force $\mathbf{Q}_{\mathrm{b}}$ is given by

$$\mathbf{Q}_{\mathrm{b}} = \mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}} = \begin{Bmatrix} 0 & 0 & 0 \\ \mathbf{I} & 0 & \mathbf{I} \\ -\mathbf{I} & 0 & \mathbf{I} \\ 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{I} \\ 0 & -\mathbf{I} & -\mathbf{I} \end{Bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \\ \boldsymbol{\lambda}_3 \end{Bmatrix} \tag{48}$$

Of several possibilities we present two realizations: maximal redundancy and minimal redundancy.

6.1. *Maximally redundant case*

If one introduce a localized interface force for every non-zero entries in (48), then one obtains

$$\mathbf{Q}_{\mathrm{b}} = \mathbf{N}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}} = \mathbf{N}_{\mathrm{b}}^{(s)}\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}, \quad \mathbf{N}_{\mathrm{b}}^{(s)} = \begin{Bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{I} & \mathbf{I} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\mathbf{I} & \mathbf{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & -\mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\mathbf{I} & -\mathbf{I} \end{Bmatrix}, \quad \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} = \begin{Bmatrix} \boldsymbol{\lambda}_1^{(1)} \\ \boldsymbol{\lambda}_2^{(1)} \\ \boldsymbol{\lambda}_1^{(2)} \\ \boldsymbol{\lambda}_2^{(2)} \\ \boldsymbol{\lambda}_1^{(3)} \\ \boldsymbol{\lambda}_2^{(3)} \\ \boldsymbol{\lambda}_1^{(4)} \\ \boldsymbol{\lambda}_2^{(4)} \end{Bmatrix} \tag{49}$$

Thus, their redundancy constraint operator $\mathbf{Z}_\lambda$ can be obtained as

$$\boldsymbol{\lambda}_{\mathrm{b}}^{(s)} = \mathbf{L}_\lambda \boldsymbol{\lambda}_{\mathrm{b}}, \quad \mathbf{L}_\lambda = \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} \\ \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix}, \quad \mathbf{Z}_\lambda = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 & \mathbf{I} \\ -\mathbf{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\mathbf{I} & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I} & -\mathbf{I} \\ 0 & -\mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\mathbf{I} & -\mathbf{I} \end{bmatrix} \tag{50}$$

It should be noted that $\mathbf{Z}_\lambda$ satisfies the following redundancy constraint condition on the localized Lagrange multipliers:

$$\mathbf{Z}_\lambda^{\mathrm{T}} \boldsymbol{\lambda}^{(s)} = \mathbf{0} \tag{18}$$

6.2. *Minimally redundant case*

If one introduce a localized interface force for each of the non-zero row in (48), then one obtains

$$\mathbf{Q}_{\mathrm{b}} = \mathbf{N}_{\mathrm{b}} \boldsymbol{\lambda}_{\mathrm{b}} = \mathbf{N}_{\mathrm{b}}^{(s)} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)}, \quad \mathbf{N}_{\mathrm{b}}^{(s)} = \begin{Bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \end{Bmatrix}, \quad \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} = \begin{Bmatrix} \boldsymbol{\lambda}^{(1)} \\ \boldsymbol{\lambda}^{(2)} \\ \boldsymbol{\lambda}^{(3)} \\ \boldsymbol{\lambda}^{(4)} \end{Bmatrix} \tag{51}$$

so that $\mathbf{L}_\lambda$ becomes $\bar{\mathbf{N}}_{\mathrm{b}}$ given by (34).

By rearranging $\mathbf{Q}_{\mathrm{b}}$ in terms of the interior and the boundary nodes, the non-trivial part of $\mathbf{Q}_{\mathrm{b}}$ can be simply written as

$$\bar{\mathbf{Q}}_{\mathrm{b}} = \mathbf{I}_{\mathrm{b}}^{(s)} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)}, \quad \mathbf{I}_{\mathrm{b}}^{(s)} = \begin{Bmatrix} \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \end{Bmatrix}, \quad \mathbf{Q}_{\mathrm{b}} = \begin{Bmatrix} \mathbf{0} \\ \bar{\mathbf{Q}}_{\mathrm{b}} \end{Bmatrix} \tag{52}$$

Therefore, the corresponding localization interface constraint operator is obtained as

$$\mathbf{Z}_\lambda = \mathbf{L}_{\mathrm{b}} = \langle \mathbf{I}\ \mathbf{I}\ \mathbf{I}\ \mathbf{I} \rangle^{\mathrm{T}} \tag{53}$$

which indicates that the localized Lagrange redundant multipliers satisfies the following constraint:

$$(\boldsymbol{\lambda}^{(1)} + \boldsymbol{\lambda}^{(2)} + \boldsymbol{\lambda}^{(3)} + \boldsymbol{\lambda}^{(4)}) = 0 \tag{54}$$

Thus, the preceding localized interface displacement operator to be an identity matrix $\mathbf{I}_{\mathrm{b}}^{(s)}$ and the Lagrange multipliers redundancy constraint operator given by the interface assembly matrix

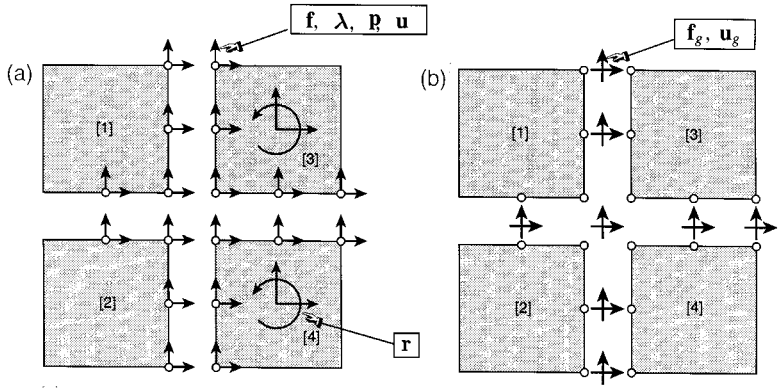

Figure 4. Minimally redundant choice and the resulting $\mathbf{I}_{\mathrm{b}}^{(s)}$ and $\mathbf{L}_{\mathrm{b}}$: (a) by assigning at each boundary nod a set of independent interface variable, the interface operator becomes $\mathbf{I}_{\mathrm{b}}^{(s)}$ as given by (52). Note that only interface nodal dofs information is all that is necessary for each subdomain; (b) the corresponding localized redundant Lagrange multipliers constraint condition is the fem assembly operator at the interface boundary given by $\mathbf{L}_{\mathrm{b}}$ perhaps the simplest choice and implementation ease

$\mathbf{L}_{\mathrm{b}}$ are uniquely determined from the finite element assembly Boolean operator $\mathbf{L}$. This simplicity constitutes a key algorithmic and implementation feature of the present algebraically partitioned FETI method as shown in Figure 4.

Substituting (52b) and (53) into the localized equation (29a), one obtains

$$\begin{bmatrix} \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{I}_{\mathrm{b}}^{(s)} & \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{R}^{(s)} & \mathbf{L}_{\mathrm{b}} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{I}_{\mathrm{b}}^{(s)} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{\mathrm{b}}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \\ \mathbf{u}_{\mathrm{gb}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)} \\ \mathbf{0} \end{Bmatrix} \tag{55}$$

From now on the above equation will be referred to as *localized* algebraically partitioned FETI equation, or simply the A-FETI equation.

7. ALGORITHMIC EQUIVALENCE BETWEEN THE GLOBAL AND LOCALIZED PARTITIONED FORMULATIONS

In the preceding sections we have shown that the simplest localization choice is to choose the localization operator $\mathbf{N}_{\mathrm{b}}^{(s)}$ to be the identity matrix corresponding to the interface degrees of freedom of each subdomain, and the interface force constraint operator $\mathbf{Z}_{\lambda}$ to be the interface nodal assembly operator $\mathbf{L}_{\mathrm{b}}$. In this section we establish that the global formulation (27) and the localized formulation (29) or (55) are *algorithmically equivalent*. By algorithmic equivalence we refer to the property that the number of iterations required to solve the system equations are the same, provided both employ the same preconditioner. To demonstrate the equivalence property, first we obtain a projection operator of the interface force constraint matrix $\mathbf{L}_{\mathrm{b}}$ given by

$$\mathbf{P}_{\ell} = \mathbf{I} - \mathbf{L}_{\mathrm{b}}(\mathbf{L}_{\mathrm{b}}^{\mathrm{T}}\mathbf{L}_{\mathrm{b}})^{-1}\mathbf{L}_{\mathrm{b}}^{\mathrm{T}}, \quad \mathbf{P}_{\ell}^{\mathrm{T}}\mathbf{L}_{\mathrm{b}} = 0 \tag{56}$$

Second, we express the localized interface force $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$ as

$$\boldsymbol{\lambda}_{\mathrm{b}}^{(s)} = \mathbf{P}_{\ell}\hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} \tag{57}$$

Introducing this and premultiplying its first row of (55) by $\mathbf{P}_\ell^{\mathrm{T}}$ we obtain

$$\begin{bmatrix} \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{R}^{(s)} & \mathbf{P}_\ell \mathbf{L}_{\mathrm{b}} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{\mathrm{b}}^{\mathrm{T}} \mathbf{P}_\ell & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \\ \mathbf{u}_{\mathrm{gb}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{f}^{(s)} \\ \mathbf{0} \end{Bmatrix} \tag{58}$$

Observe that, by (56b), the solution matrix of the above equation is reduced to

$$\begin{bmatrix} \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{R}^{(s)} & \mathbf{P}_\ell \mathbf{L}_{\mathrm{b}} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{\mathrm{b}}^{\mathrm{T}} \mathbf{P}_\ell & \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_\ell \mathbf{N}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{R}^{(s)} & \mathbf{0} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{59}$$

Therefore, (58) reduces to

$$\begin{bmatrix} \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{R}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \end{Bmatrix} = \begin{Bmatrix} \mathbf{P}_\ell \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}} \mathbf{F}^{(s)} \mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}} \mathbf{f}^{(s)} \end{Bmatrix} \tag{60}$$

Comparing equation (60) with the global formulation (27), they would be *algorithmically equivalent* if the following relations hold:

$$\mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell = \mathbf{N}_{\mathrm{b}} \mathbf{N}_{\mathrm{b}}^{\mathrm{T}} \;\Leftrightarrow\; \hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} = \mathbf{N}_{\mathrm{b}}^{\mathrm{T}} \boldsymbol{\lambda}_{\mathrm{b}} \tag{61}$$

When one employs the *minimally redundant* localization operator, the above equivalence relations hold. As an example, let us take the four-subdomain problem shown in Figure 1. For this case, we have from (53) and (56)

$$\mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell = \frac{1}{4} \begin{bmatrix} 3\mathbf{I} & -\mathbf{I} & -\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & 3\mathbf{I} & -\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & -\mathbf{I} & 3\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & -\mathbf{I} & -\mathbf{I} & 3\mathbf{I} \end{bmatrix} = \mathbf{N}_{\mathrm{b}} \mathbf{N}_{\mathrm{b}}^{\mathrm{T}}, \quad \mathbf{N}_{\mathrm{b}} = \begin{bmatrix} \mathbf{I}/\sqrt{2} & 0 & \mathbf{I}/2 \\ -\mathbf{I}/\sqrt{2} & 0 & \mathbf{I}/2 \\ 0 & \mathbf{I}/\sqrt{2} & -\mathbf{I}/2 \\ 0 & -\mathbf{I}/\sqrt{2} & -\mathbf{I}/2 \end{bmatrix} \tag{62}$$

Observe that $\mathbf{N}_{\mathrm{b}}$ given by (62b) is the orthonormalized, compacted form of (48). Clearly, then we have by using (61) and (62):

$$\mathbf{I}_{\mathrm{b}}^{(s)} \mathbf{P}_\ell \hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} = \begin{bmatrix} \mathbf{I}/\sqrt{2} & 0 & \mathbf{I}/2 \\ -\mathbf{I}/\sqrt{2} & 0 & \mathbf{I}/2 \\ 0 & \mathbf{I}/\sqrt{2} & -\mathbf{I}/2 \\ 0 & -\mathbf{I}/\sqrt{2} & -\mathbf{I}/2 \end{bmatrix} \hat{\boldsymbol{\lambda}}_{\mathrm{b}}^{(s)} = \mathbf{N}_{\mathrm{b}} \boldsymbol{\lambda}_{\mathrm{b}} \tag{63}$$

so that the equivalence relations (60) and (27) are seen to hold.

Remark 3. The equivalence established between the global formulation (27) and the localized formulation (55) of the present algebraically partitioned method provides an important property. That is, it is not necessary to evaluate the algorithmic performance on parallel computers as far as iterative convergence properties are concerned. The interprocessor communications requirements

can then be estimated before the localized formulation (27) is implemented, thus facilitating algorithmic evaluations. In fact, this is the approach we have taken in the companion paper.[8]

Remark 4. From the present *algebraically partitioned formalism*, there exists no apparent justification to infuse additional modes, except the redundant interface force constraint operator $\mathbf{Z}_\lambda = \mathbf{L}_\mathrm{b}$ (19) and (53). This suggests that different localizations may lead to different localization constraints that need to be augmented to the interface problems. However, we have not been able to systematize the corner modes of Farhat and Mandel[9] within the context of the present formalism.

8. COMPARISON OF THE PRESENT FORMULATION WITH THE DIFFERENTIALLY PARTITIONED FETI METHOD

The starting point of the present derivation is the globally assembled finite element equation for structures. As with any work which has rich and fast-evolving heritage, the differentially partitioned FETI (D-FETI) method has been undergoing a rapid maturity. Therefore, it would be instructive for us to offer similarities as well as differences as best as we could have done. Our comparison is summarized in Table I below.

Based on the above summary table, we offer the following comments:

1. To the best of our knowledge, the construction of $\mathbf{B}^{(s)}$ matrix is not uniquely determined. We note that this may offer added flexibility in the differentially partitioned FETI (D-CFETI) method as to *tailor* it for different situations. On the other hand, once the structure is discretized, $\mathbf{L}$ matrix is fixed once for all. While this may not offer any flexibility, this does offer a fixed reference matrix from which all of the present kinematic interface constraint matrices are deduced.
2. We have employed the classical static equilibrium operator as the basis for obtaining the floating modes, thus by-passing the computations of nullspace matrices for each floating subdomain. It should be noted that the static equilibrium operator thus derived must provide dual roles of being the rigid-body modes and at the same time assuring that the individual floating substructure satisfies its equilibrium condition under given interface Lagrange multipliers.
3. The solution matrices of the present method, viz., the matrices in the left-hand sides of (27) and (55), possess their full rank. On the other hand, the solution matrix in the differentially partitioned FETI (D-FETI) method given by (8) becomes rank-deficient whenever any subdomain has a cross point.

Table I. Comparison of D-FETI with present A-FETI method

Attributes	D-FETI	Present A-FETI method
Interface condition	$\sum^{n_s} \mathbf{B}^{(s)}\mathbf{u}^{(s)} = 0$ (2)	$\mathbf{u}^{(s)} - \mathbf{L}\mathbf{u}_\mathrm{g} = 0$ (11)
Floating modes	$\begin{bmatrix} -\mathbf{K}_{cc}^{-1}\mathbf{K}_{cr} \\ \mathbf{I} \end{bmatrix}$ (39)	$\begin{bmatrix} \mathbf{I} & \mathbf{0} & \dots \\ \boldsymbol{\chi} & \mathbf{I} & \dots \end{bmatrix}$ (40)–(47)
Localization constraint	??	$\mathbf{I}_\mathrm{b}^\mathrm{T}\mathbf{u}_\mathrm{b}^{(s)} - \mathbf{L}_\mathrm{b}\mathbf{u}_\mathrm{gb} = 0$ (18)
Corner modes	$\mathbf{C}^\mathrm{T} \sum^{N_s} \mathbf{B}^{(s)}\mathbf{u}^{(s)} = 0$ (30)	??
Solution matrix	Generally rank-deficient	Full rank

4. Regarding the localization constraint condition (18) of the present method and the corner modes of the differentially partitioned FETI2,[9, 10] we offer the following discussion:

In terms of the saddle-point formulation, the added corner mode constraint quoted in (30) from Farhat and Mandel[9] and Mandel *et al.*[10] would take the following functional form:*

$$J_{\mathrm{DFETI}} = \mathbf{u}^{(s)^{\mathrm{T}}}(\mathbf{f}^{(s)} - \tfrac{1}{2}\mathbf{K}^{(s)}\mathbf{u}^{(s)}) - \boldsymbol{\lambda}_{\mathrm{b}}^{(s)^{\mathrm{T}}}(\mathbf{B}^{(s)}\mathbf{u}^{(s)}) - \boldsymbol{\lambda}_{c}^{(s)^{\mathrm{T}}}(\mathbf{C}^{\mathrm{T}}\mathbf{B}^{(s)}\mathbf{u}^{(s)}) \tag{64}$$

Stationarity of the above functional would lead to the following set of equations:

$$\begin{aligned} \mathbf{K}^{(s)}\mathbf{u}^{(s)} &= \mathbf{f}^{(s)} - \mathbf{B}^{(s)^{\mathrm{T}}}\boldsymbol{\lambda}_{\mathrm{b}}^{(s)} - \mathbf{B}^{(s)^{\mathrm{T}}}\mathbf{C}\boldsymbol{\lambda}_{c}^{(s)} \\ \mathbf{B}^{(s)}\mathbf{u}^{(s)} &= 0 \\ \mathbf{C}^{\mathrm{T}}\mathbf{B}^{(s)}\mathbf{u}^{(s)} &= 0 \end{aligned} \tag{65}$$

which leads to the following system of equations:

$$\begin{bmatrix} \mathbf{F}_I^{(s)} & \mathbf{F}_I^{(s)}\mathbf{C} & \mathbf{G}^{(s)} \\ \mathbf{C}^{\mathrm{T}}\mathbf{F}_I^{(s)} & \mathbf{C}^{\mathrm{T}}\mathbf{F}_I^{(s)}\mathbf{C} & \mathbf{C}^{\mathrm{T}}\mathbf{G}^{(s)} \\ \mathbf{G}^{(s)^{\mathrm{T}}} & \mathbf{G}^{(s)^{\mathrm{T}}}\mathbf{C} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{c}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \end{Bmatrix} = \begin{Bmatrix} \mathbf{B}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{K}^{(s)^{+}}\mathbf{f}^{(s)} \\ \mathbf{C}^{\mathrm{T}}\mathbf{B}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{K}^{(s)^{+}}\mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)} \end{Bmatrix} \tag{66}$$

$$\mathbf{F}_I^{(s)} = \mathbf{B}^{(s)}\mathbf{K}^{(s)^{+}}\mathbf{B}^{(s)^{\mathrm{T}}}, \quad \mathbf{G}^{(s)} = \mathbf{R}^{(s)}\mathbf{B}^{(s)^{\mathrm{T}}}$$

We note that the subdomain flexibility $\mathbf{F}_I^{(s)}$ must be utilized in computing $\boldsymbol{\lambda}_{\mathrm{c}}^{(s)}$. In addition, the solution of $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$ must account for the coupling matrix $\mathbf{F}_I^{(s)}\mathbf{C}$ in the solution matrix.
The present localized saddle-point functional is given by

$$J_{\mathrm{AFETI}} = \mathbf{u}^{(s)^{\mathrm{T}}}(\mathbf{f}^{(s)} - \tfrac{1}{2}\mathbf{K}^{(s)}\mathbf{u}^{(s)}) - \boldsymbol{\lambda}_{\mathrm{b}}^{(s)^{\mathrm{T}}}(\mathbf{I}_{\mathrm{b}}^{\mathrm{T}}\mathbf{u}_{\mathrm{b}}^{(s)} - \mathbf{Z}_{\lambda}\mathbf{u}_{\mathrm{gb}}) \tag{67}$$

which leads to (29) and (55) as reproduced here for comparison clarity:

$$\begin{bmatrix} \mathbf{F}_{\mathrm{b}}^{(s)} & \mathbf{R}_{\mathrm{b}}^{(s)} & \mathbf{L}_{\mathrm{b}} \\ \mathbf{R}_{\mathrm{b}}^{(s)^{\mathrm{T}}} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{\mathrm{b}}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\lambda}_{\mathrm{b}}^{(s)} \\ \boldsymbol{\lambda}_{\mathrm{r}}^{(s)} \\ \mathbf{u}_{\mathrm{gb}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{f}^{(s)} \\ \mathbf{R}^{(s)^{\mathrm{T}}}\mathbf{f}^{(s)} \\ \mathbf{0} \end{Bmatrix}, \quad \begin{aligned} \mathbf{F}_{\mathrm{b}}^{(s)} &= \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{F}^{(s)}\mathbf{I}_{\mathrm{b}}^{(s)} \\ \mathbf{R}_{\mathrm{b}}^{(s)} &= \mathbf{I}_{\mathrm{b}}^{(s)^{\mathrm{T}}}\mathbf{R}^{(s)} \end{aligned} \tag{68}$$

Observe that, if one views the interface global displacement $\mathbf{u}_{\mathrm{gb}}$ as additional Lagrange multipliers, it is coupled to $\boldsymbol{\lambda}_{\mathrm{b}}^{(s)}$ only through the kinematic operator $\mathbf{L}_{\mathrm{b}}$, thus not involving any subdomain flexibility $\mathbf{F}_{\mathrm{b}}^{(s)}$.

Thus, the crucial difference is that the differentially partitioned FETI (D-FETI) method must introduce the undetermined Lagrange multipliers $\boldsymbol{\lambda}_{\mathrm{c}}^{(s)}$ in order to enforce the constraint condition (65c). In contrast, the present formulation (A-FETI) utilizes $\mathbf{u}_{\mathrm{gb}}$ as an independent unknown, namely, the global displacement vector at the domain interfaces. This is a consequence of utilizing the well-known finite element assembly equation (11), we believe.

*The saddle-point functional we adopt herein is not identical to the one utilized in Farhat and Mandel. Specifically, the corner modes constraint is differently incorporated. Hence, this may add an alternative way of implementing the corner modes

9. DISCUSSIONS

An algebraically partitioned FETI method is presented for the solution of large-scale computational mechanics problems on parallel computers. The present method enjoys the following features:

The method yields a full-rank solution matrix with or without the cross subdomain interfacing points for both the global interface formulation (27) and the localized interface formulation (29) or (55) or (68). In particular, for the localized formulation (68), the use of the interface force constraint condition (18) satisfies its rank-sufficiency requirement.

The global subdomain displacement compatibility operator, $\mathbf{N}_b$ or localized $\mathbf{I}_b$, can be explicitly constructed using only the partitioned subdomain connectivity topology.

Likewise, the subdomain force equilibrium operator $\mathbf{R}^{(s)}$ can be explicitly constructed using only the finite element mesh geometries, thus by-passing the use of subdomain stiffness matrices which can be error prone for complex subdomain meshes. The resulting matrix is independent of system non-linearities and element embelishments. One important application of the present subdomain force equilibrium operator $\mathbf{R}^{(s)}$ or floating modes is in the development of a so-called *matrix-free FETI algorithm* for a class of highly nonlinear problems. For these problems, as the tangent stiffness matrices are not available, the present approach is perhaps the only way to compute the subdomain floating modes. This algorithm is currently under active development by the authors.

The use of the subdomain force equilibrium condition (23) or (25) sheds physical insight into the importance of the high-accuracy requirement of the subdomain force equilibrium operator $\mathbf{R}^{(s)}$. In viewing $\mathbf{R}^{(s)}$ as subdomain rigid-body modes, its role is to kinematically patch the floating domains together. In the present view, it not only patches up the floating subdomains together, but more importantly must satisfy the subdomain equilibrium.

The constraint condition on the localized interface Lagrange multipliers given by (18) manifests naturally from the domain-by-domain vs. global displacement relation given by (11), a fundamental feature of the present algebraically partitioned FETI method. This and other features of the present method compared with the differentially partitioned FETI method are summarized in Table I in Section 8.

It has been shown that the global and localized formulations given by (26) and (28) are algorithmically equivalent. This means that, as long as they employ the same preconditioner, their iterative efficiency would be the same. Their major differences are in their respective interprocessor communication needs.

The solution matrices of the present method both in the global formulation (27) and the localized formulation (29) retain their full ranks. On the other hand, the solution matrix in the differentially partitioned FETI method given by (8) becomes rank-deficient whenever any subdomain has a cross point. This may offer additional solution strategies for the present algebraically partitioned formalism.

While the differentially partitioned FETI method with corner modes as given by (65) may be computationally more intensive than the present algebraically partitioned FETI method (67), the former may offer commensurate computational efficiency. Undoubtedly, their differences would lead to different computer implementations and consequently differing computational performance. Both the implementational ease and the iterative efficiency of the present method is not well evaluated, however. We will report on the computer implementation details and some preliminary numerical evaluations of the present algebraically partitioned FETI method in a companion paper.[8]

ACKNOWLEDGEMENTS

This research has been partially funded by NSF/HPCC Grant ASC-9217394 and by Sandia National Laboratories under an ASCI Initiative seed contract. It is a pleasure to offer our thanks to our

colleague Prof. Charbel Farhat for his unending enthusiasm and many helpful discussions. We also thank Prof. Patrick Le Tallec for his constructive comments on the static equilibrium operator construction details. The second author was supported by a post-doctoral fellowship from the National Council for Research and Development (CNPq-Brazil).

REFERENCES

1. C. Farhat and F.-X. Roux, 'A method of finite element tearing and interconnecting and its parallel solution algorithm', *Int. j. numer. methods eng.*, **32,** 1205–1227 (1991).
2. P. LeTallec, 'Domain decomposition methods in computational mechanics', *Comput. Mech. Adv.*, 121–220 (1994).
3. C. Farhat and F.-X. Roux, 'Implicit parallel processing in structural mechanics', *Comput. Mech. Adv.*, **2,** 1–124, Amsterdam (1994).
4. C. A. Felippa and K. C. Park, 'Staggered transient analysis procedures for coupled mechanical systems', *Comput. Methods Appl. Mech. Eng.*, **24,** 61–111 (1980).
5. K. C. Park, 'Partitioned transient analysis procedures for coupled-field problems: Stability analysis', *J. Appl. Mech.*, **47,** 370–376 (1980).
6. K. C. Park and C. A. Felippa, 'Partitioned analysis of coupled systems', in: T. Belytschko and T. J. R. Hughes (eds.), *Computational Methods for Transient Analysis*, Elsevier, Amsterdam, 1983, pp. 157–219.
7. K. C. Park and C. A. Felippa, 'Recent developments in coupled-field analysis methods', in R. W. Lewis *et al.* (eds.), *Numerical Methods in Coupled Systems*, Wiley, New York, 1984, pp. 327–352.
8. M. R. Justino, Jr., K. C. Park and C. A. Felippa, 'An algebraically partitioned FETI method for parallel structural analysis: implementation', *Report No. CU-CAS-96-10*, Center for Aerospace Structures, University of Colorado, 1996.
9. C. Farhat and J. Mandel, 'The two-level FETI method for static and dynamic plate problems—Part I: an optimal iterative solver for biharmonic systems', *Report Number CU-CAS-95-23*, Center for Aerospace Structures, University of Colorado, Boulder, CO, 1995.
10. J. Mandel, R. Tezaur and C. Farhat, 'An optimal Lagrange multiplier based domain decomposition method for plate bending problems', *Report Number CU-CAS-95-15*, Center for Aerospace Structures, University of Colorado, Boulder, CO, 1995.

